Distribution Agreement

In presenting this thesis or dissertation as a partial fulfillment of the requirements for an advanced degree from Emory University, I hereby grant to Emory University and its agents the non-exclusive license to archive, make accessible, and display my thesis or dissertation in whole or in part in all forms of media, now or hereafter known, including display on the world wide web. I understand that I may select some access restrictions as part of the online submission of this thesis or dissertation. I retain all ownership rights to the copyright of the thesis or dissertation. I also retain the right to use in future works (such as articles or books) all or part of this thesis or dissertation.

Signature:

____________________________________ ____________

Sijian Fan Date

Improved Algorithm for Independent Component Analysis (ICA) with the Relax and Split Approximation

By

Sijian Fan
Master of Science in Public Health

Biostatistics (BIOS) and Bioinformatics

Benjamin B. Risk, Ph.D.
Thesis Advisor

Howard Chang, Ph.D.
Thesis Reader

Improved Algorithm for Independent Component Analysis (ICA) with the Relax and Split Approximation

By

Sijian Fan
B.A., Zhejiang University, China, 2018

Thesis Advisor: Benjamin B. Risk, Ph.D.

An abstract of
A thesis submitted to the Faculty of the
Rollins School of Public Health of Emory University
in partial fulfillment of the requirements for the degree of
Master of Science in Public Health
in Biostatistics (BIOS) and Bioinformatics
2020

Abstract

Improved Algorithm for Independent Component Analysis (ICA) with the Relax and Split Approximation
By Sijian Fan

Independent component analysis (ICA) has been increasingly used to separate sources and extract features in signal processing and neuroimaging studies. To overcome its computational problems with local optima, as well as problems with non-smooth and non-convex objective functions, relax and split optimization was applied in this study and comparisons were made between the refined algorithms and the popular FastICA algorithm. A tuning parameter was used to control the relaxation and sparsity level of the Relax-Laplace method (with an objective function derived from the Laplace density), and to control the relaxation level of the Relax-logistic method (with an objective function derived from the logistic density).

We conducted a simulation study to examine the impact of the tuning parameter on accuracy and sensitivity to initialization. We found smaller values of the tuning parameter can lead to accurate estimates of the components while having fewer issues with local optima relative to FastICA, while larger values can result in inaccuracies. Running 1000 times with a pool of 50 initializations, we found Relax-Laplace algorithm was the most accurate and consistent one compared with Relax-logistic, FastICA-logistic, and FastICA-tanh.

We conducted a multi-subject analysis of functional magnetic resonance imaging (fMRI) data from the Human Connectom Project using Relax-Laplace, FastICA-logistic, and FastICA-tanh. In a pool of 50 initializations, the Relax-Laplace returns the same result for all initializations, whereas both FastICA-logistic and FastICA-tanh converged on the estimate of the argmax in just over half of the initializations. Moreover, the Relax-Laplace produced sparse figures for the rs-fMRI data that highlight features of resting-state networks.

Improved Algorithm for Independent Component Analysis (ICA) with the Relax and Split Approximation

By

Sijian Fan
B.A., Zhejiang University, China, 2018

Thesis Advisor: Benjamin B. Risk, Ph.D.

A thesis submitted to the Faculty of the
Rollins School of Public Health of Emory University
in partial fulfillment of the requirements for the degree of
Master of Science in Public Health
in Biostatistics (BIOS) and Bioinformatics
2020

Acknowledgements

Data were provided [in part] by the Human Connectome Project, WU-Minn Consortium (Principal Investigators: David Van Essen and Kamil Ugurbil; 1U54MH091657) funded by the 16 NIH Institutes and Centers that support the NIH Blueprint for Neuroscience Research; and by the McDonnell Center for Systems Neuroscience at Washington University. Thanks a lot for their provision of data and their permission of using data for my thesis.

I also thank sincerely for the concerns and guidance from Benjamin B. Risk during the whole period of this study. Thank you to Howard Chang for sharing his knowledge on writing my thesis. Also many thanks to Peng Zheng, Sasha Aravkin, and Irina Gaynanova for their help and their formulation of the algorithms used in this thesis. Last but not least, I thank my family and friends for their concerns and encouragement towards my thesis.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

Independent component analysis (ICA) is a method for blind source separation commonly used in signal processing and neuroimaging. The goal of ICA is to find a linear transformation of the original data to recover independent components, or make the components as independent as possible [7]. Two dominant applications of ICA are separating sources and extracting features [6]. In cognitive neuroscience, ICA is a popular approach to separate brain signals, and it is commonly applied to functional magnetic resonance imaging (fMRI) [1]. Resting-state fMRI (rs-fMRI) is measured when participants are assigned without particular task, wherein spontaneous fluctuations in brain activity are measured across time [2]. Group ICA can be applied to multi-subject resting-state fMRI data to estimate independent components. The independent components correspond to resting-state "networks," or regions of the brain that tend to spontaneously co-activate [3].

ICA is a challenging computational problem because it commonly involves non-convex optimization, which arises due to orthogonality constraints and can be made worse by the choice of non-linearity in the objective function. In practice, different measurements of independence are used in different ICA methods, such as minimum mutual information (MI), maximum likelihood (ML), and negentropy [4]. Actually, these measurements are related with each other: under the constraint of orthogonal matrices, the problem of minimizing MI will be equivalent to the problem of maximizing ML, and also be equal to the measurement of negentropy. The very popular FastICA method applied an approximation on the measurement of negentropy, where log hyperbolic cosine is generally used as the contrast function during approximation [6]. Because it works very fast even for a large dataset, FastICA method has been used in fMRI analysis. Evaluations has already been made to assess its performance on applications to rs-fMRI data [9]. In the evaluation, the issue of local optima appeared even when only two components were mixed, like the asymmetric mixture of two normal densities. Situations would become worse if using more complex non-convex functions. Local optima can produce dramatically different estimators in both simulations and fMRI applications. To deal with this problem, starting with multiple initialization is not always a valid option due to the large computation expense, especially when dealing with big data-set.

In real practice, principle component analysis (PCA) is often used for data whitening and pre-processing before ICA, and the dimension reduction can help to save a lot of computation time. However, PCA may remove important information due to the reduction of dimensions, which can no longer be recovered. Recently, methods like linear non-Gaussian component analysis (LNGCA) was proposed to deal with such situations [8]. This method formulated a linear non-Gaussian component model with Gaussian noise component, maximum likelihood was used to represent the most important information in the data, rather than the variance in PCA. However, this method has even more problems with local optima, further improvements are needed to get a better performance of ICA.

The goals of this study are two-fold: 1) propose algorithms that are less sensitive to initialization, thus reducing the computational challenges arising from performing ICA algorithms with hundreds of initial values; 2) utilize the Laplace distribution in the objective function, which results in a sparse ICA. Historically, sparse ICA was a challenging problem due to the non-smoothness of the Laplace density, which prevents the use of the classic FastICA algorithm involving an approximative Newton step. Zheng et al. developed the relax-and-split algorithm for sparse ICA, which they implemented in the Julia programming language [10, 11] . Here we translate their algorithm to the R programming language and conduct a novel simulation study.

In Section 2, we will introduce briefly about the methods used in this study, mainly focusing on the general structures of ICA, LNGCA and the relax and split method to improve existing ICA algorithms. Simulation and applications on real data will also be introduced. In Section 3, we will talk about the results generated by simulation and apply the relax-and-split algorithm to Group ICA of rs-fMRI from the Human Connectome Project. In Section 4, we will discuss the findings of this study, as well as the restrictions or limitations of our methods.

Chapter 2

Method

2.1 Classic ICA and Linear Non-Gaussian Component Analysis (LNGCA)

The typical expression of classic ICA algorithm is $\mathbf{X} = \mathbf{MS}$, where $\mathbf{X} \in \mathbb{R}^m$ is a random vector, $\mathbf{S} \in \mathbb{R}^n$ is the independent components random vector with mutually independent elements, and m and n are usually assumed to be equal so that $\mathbf{M} \in \mathbb{R}^{m \times n}$ is a square matrix.

To allow dimension reduction in ICA, Linear Non-Gaussian Component Analysis (LNGCA) was proposed using the concepts of signal and noise [8]. It expanded the classic ICA expression as the following:

$$\mathbf{X} = \mathbf{M_S S} + \mathbf{M_N N} \tag{2.1}$$

where $\mathbf{X} \in \mathbb{R}^m$ is a random vector, $\mathbf{S} \in \mathbb{R}^n$ is the independent components vector, and $\mathbf{N} \in \mathbb{R}^{m-n}$ is the noise vector. Correspondingly, we can show the dimension of $\mathbf{M_S} \in \mathbb{R}^{m \times n}$, $\mathbf{M_N} \in \mathbb{R}^{m \times (m-n)}$, and therefore $\mathbf{M} = [\mathbf{M_S}, \mathbf{M_N}]$ (by concatenation) will be full rank.

The classic ICA can be viewed as a noise-free version under this expression for LNGCA, namely $n = m$ or $m - 1$. One Gaussian component is allowed here because the last component can be derived from others. For identifiability, we assume $\mathrm{E}(\mathbf{S}) = \mathbf{0}$, $\mathrm{E}(\mathbf{N}) = \mathbf{0}$, and $\mathrm{E}(\mathbf{S}\mathbf{S}^T) = I_n$, namely the data is centered by their mean to have zero mean and unit variance. The target of ICA is to estimate the unmixing matrix $\mathbf{W} = \mathbf{M}^{-1}$. Then $\mathbf{W}$ is identifiable up to sign and permutation of the columns. We will introduce the methods briefly below.

2.2 Entropy and Mutual Information

In signal processing, entropy is used to measure the quantity of information. Here we apply this concept to measure the information of vectors after unmixing by ICA. Denote $H(\mathbf{y})$ as the entropy of a given vector $\mathbf{y} = [y_1, y_2, ..., y_n]^T$ with density function of $f(.)$,

$$H(\mathbf{y}) = -\int f(\mathbf{y}) \log f(\mathbf{y}) d\mathbf{y} \tag{2.2}$$

Then we can define the mutual information (MI) between random variables X and Y:

$$MI = I(X;Y) = H(X) - H(X|Y) \tag{2.3}$$

Suppose we have a classic ICA model with $\mathbf{S} \in \mathbb{R}^m$ as the signal vector, then its mutual information will be defined as the sum of marginal entropy minus its joint entropy [7]:

$$\begin{aligned} MI = I(S_1; S_2; ...; S_m) &= \sum_{i=1}^{m} H(S_i) - H(\mathbf{S}) \\ &= -\int \log\left(\prod_{i=1}^{m} f_{S_i}(S_i)\right) f_{\mathbf{S}}(\mathbf{S}) d\mathbf{S} + \int \log\left(f_{\mathbf{S}}(\mathbf{S})\right) f_{\mathbf{S}}(\mathbf{S}) d\mathbf{S} \\ &= \int \log\left(\frac{f_{\mathbf{S}}(\mathbf{S})}{\prod_{i=1}^{m} f_{S_i}(S_i)}\right) f_{\mathbf{S}}(\mathbf{S}) d\mathbf{S} \end{aligned} \tag{2.4}$$

where $f_{\mathbf{S}}(\mathbf{S})$ denotes the joint density function of $\mathbf{S}$, and $f_{S_i}(S_i)$ denotes the marginal density function of ith component in $\mathbf{S}$. Let $F_{\mathbf{S}}$ as the distribution of $\mathbf{S}$, let F_{S_i} as the marginal distribution of S_i, we can have $F_{\mathbf{S}} = \prod_{i=1}^{m} F_{S_i}$ if the vector $\mathbf{S}$ contains truly independent components, and the mutual information will be exactly zero.

For the ICA model of $\mathbf{X} = \mathbf{MS}$ with k identically distributed samples $\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_k$, we examine among the set $\mathbf{U}$ ($\mathbf{U} \in \mathbb{R}^{m\times m}$) to find the true unmixing matrix $\mathbf{W}$ ($\mathbf{W} \in \mathbb{R}^{m\times m}$), where $\mathbf{U}$ is orthogonal matrices after data pre-whitening. In practice, the $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ are not independent, but the objective functions can still successfully unmix signals. In spatial ICA of fMRI, k correspond to the number of voxels, or volumetric pixels. Let $\mathbf{S}$ be the truly independent signal vector, we can define the objective function from equation (4) as

$$L_{MI}(\mathbf{U}) = -\sum_{j=1}^{k} \sum_{i=1}^{m} \log f_{S_i}(\boldsymbol{u}_i' \boldsymbol{x}_j) + \sum_{j=1}^{k} \log f_{\mathbf{S}}(\mathbf{U}\boldsymbol{x}_j) \tag{2.5}$$

where $\boldsymbol{u}_i'$ is used to represent ith row of $\mathbf{U}$ matrix to match the column vectors ($\boldsymbol{x}_j$). Since the second part is invariant to rotations of $\mathbf{U}$ under the orthogonal assumption, the problem in ICA becomes:

$$\hat{\mathbf{W}} = \arg\min_{\mathbf{U}} -\sum_{j=1}^{k} \sum_{i=1}^{m} \log f_{S_i}(\boldsymbol{u}_i' \boldsymbol{x}_j) \tag{2.6}$$

The above equation is equivalent to maximize the maximum likelihood for the observed data $\boldsymbol{x}_1, ..., \boldsymbol{x}_k$, namely the criterion of minimum mutual information is equal to the criterion of maximum likelihood under the orthogonal assumption.

2.3 Negentropy and Approximation for FastICA Algorithm

Negentropy is a normalized entropy but with better properties, it is invariant under linear transformations. Based on the equation (2) in above and a vector $\mathbf{y} = [y_1, y_2, ..., y_n]^T$, the negentropy can be expressed as the following:

$$J(\mathbf{y}) = H(\mathbf{y}_{Gauss}) - H(\mathbf{y}) \tag{2.7}$$

where $\mathbf{y}_{Gauss} \sim N(0, E(\mathbf{y}\mathbf{y}^T))$ is a Gaussian distribution with the same covariance matrix with $\mathbf{y}$. Thus, negentropy can be used as a measurement for non-Gaussianity [6], which is very helpful to explain mutual information. Similar to the equation (4), under the ICA structure of $\mathbf{X} = \mathbf{MS}$, the mutual information can be expressed as

$$MI = I(S_i; S_2; ...; S_m) = J(\mathbf{UX}) - \sum_{i=1}^{m} J(\boldsymbol{u}_i'\mathbf{X}) \tag{2.8}$$

where $\mathbf{U}$ is an estimator for the true unmixing matrix $\mathbf{W}$ as mentioned above.

FastICA was then develpoed based on the negentropy with the approximation of

$$J(\mathbf{X}) \approx c[E\{G(\mathbf{X})\} - E\{G(\mathbf{Y})\}]^2 \tag{2.9}$$

where c is a constant, and G is a nonquadratic function with common choices of

$$
\begin{aligned}
G_1(y) &= \frac{1}{a_1}\log(cosh(a_1 y)) \\
g_1(y) &= tanh(a_1 y) \\
G_2(y) &= \log(1 + e^{y \cdot a}) \\
g_2(y) &= \frac{a \cdot e^{y \cdot a}}{1 + e^{y \cdot a}}
\end{aligned}
$$

The derivative is given for the future use in the following parts. And now we can have a new objective function for ICA:

$$L_{FastICA}(\mathbf{U}) = \sum_{i=1}^{m} \left[\frac{1}{m} \sum_{j=1}^{k} E\{G(\boldsymbol{u}_i' \boldsymbol{x}_j)\} - E\{G(\mathbf{Y})\} \right]^2 \tag{2.10}$$

where E{G(Y)} is a constant. This objective function can be maximized by using an approximative Newton's step in a fixed-point algorithm [6]. For the FastICA method, both “log cosh” (FastICA tanh) and “logistic” (FastICA logistic) were applied in this study. Note the logistic function is used in the Infomax algorithm, which uses gradient descent, but here we use the FastICA algorithm for the logistic objective function.

2.4 Relax and Split Optimization for ICA

The idea of relax and split optimization was proposed by Peng Zheng and Aleksandr Aravkin [10] for nonsmooth and nonconvex problems. Here we apply this algorithm to deal with non-smooth objective functions like Laplace density, which can greatly improve the performance of ICA, especially when we observed high dimensional data. Sparse components will be estimated under the Laplace density (Sparse ICA). Besides, since relax and split algorithm is also compatible with smooth functions, we also applied it on the objective function derived from logistic density. Other extremely irregular (nonconvex) function G can also use the similar procedure to reduce the sensitivity to initialization values. Below we introduce the general structure of using relax and split optimization.

Based on the goal of minimizing mutual information, we will arrive at the problem:

$$\min_{\mathbf{U}} f(\mathbf{UX}), s.t. \mathbf{U}^T\mathbf{U} = \mathbf{I}, \tag{2.11}$$

where $\mathbf{X} \in \mathbb{R}^{m \times k}$ is the whitened observed data matrix, $\mathbf{U}$ is the orthogonal transformation matrix that estimates the true unmixing matrix $\mathbf{W}$, and $f(\mathbf{X}) = \sum_i \sum_j G(x_{ij})$. Using the relax and split problem class, we can express the original problem under a structure like:

$$\min_{x} h(Ax) + g(x), \tag{2.12}$$

where $h(.)$ can be nonsmooth and nonconvex functions, and $g(.)$ is a quadratic function. Here, we introduce an auxiliary variable $\mathbf{V}$ and consider the relaxed objective, the above question can be written as the following:

$$\arg\min_{\mathbf{U},\mathbf{V}} \; f(\mathbf{V}) + \frac{1}{2\nu}\|\mathbf{V} - \mathbf{UX}\|_F^2, s.t. \mathbf{U}^T\mathbf{U} = I, \tag{2.13}$$

where the parameter ν is a tuning parameter whose impact we evaluate in Result 3.1. The general steps are summarized in Algorithm 1.

The choice of $f(.)$ in (2.13) will impact which optimization can be used in the sub-problem 2(ii) in Algorithm 1. When updating $\mathbf{V}$, we considered the Laplace (i.e., double exponential) density, which has a closed form solution (Part 2.5), and the logistic function, which can be optimized using Newton's method (Part 2.6). The Laplace density is also used in the Lasso, and can also be considered an L1 penalty on the magnitude of the elements of the independent components. The closed form

```
Algorithm 1: Relax and Split Algorithm for ICA
  Inputs : $\mathbf{V}^{(0)}$, $\nu$, whitened data $\mathbf{X}$
  Output: $\mathbf{U}^{(n)}, \mathbf{V}^{(n)}$
    1. Initialize: $k$=0 and $\mathbf{V}^{(0)}$ is equal to the first three principle components
    2. While not converged
          (i). $\mathbf{U}^{(k+1)} \leftarrow \arg\min_{\mathbf{U}^T\mathbf{U}=\mathbf{I}} \|\mathbf{V}^{(k)} - \mathbf{U}\mathbf{X}\|_F^2$
          (ii). $\mathbf{V}^{(k+1)} \leftarrow \arg\min_{\mathbf{V}} f(\mathbf{V}) + \frac{1}{2\nu}\|\mathbf{V} - \mathbf{U}^{(k+1)}\mathbf{X}\|_F^2$
          (iii). $k \leftarrow k + 1$
```

solution is easy to apply for a big data with simple coding structures. The logistic density is also used in InfomaxICA. Here we use Newton's method in combination with the relax-and-split framework, which is to very useful to deal with smooth G functions.

Updating $\mathbf{U}$ is equivalent to the orthogonal Procrustes problem, which is solved using a singular vector decomposition (SVD), see Part 2.7.

2.5 Solution for Laplace Density

Independent components can be estimated by assuming the components have a Laplace density. Sparse components will be produced under the Laplace density:

$$f(x) = \frac{1}{2\lambda} e^{-\frac{|x-\mu|}{\lambda}} \tag{2.14}$$

where μ here is equal to 0 since we have centered the data. The variance is $2\lambda^2$, and therefore we set $\lambda = \frac{\sqrt{2}}{2}$ to make the default situation with unit variance.

To start up, we first express the ICA algorithm as the goal of minimizing mutual information, which eventually is the same as maximizing its log likelihood function as mentioned at equation (6):

$$\begin{aligned}\hat{\mathbf{W}} &= \arg\min_{\mathbf{U}} - \sum_{j=1}^{k}\sum_{i=1}^{m} \log f_{S_i}(\boldsymbol{u}_i'\boldsymbol{x}_j) \\ &= \arg\min_{\mathbf{U}} - \sum_{j=1}^{k}\sum_{i=1}^{m} G(\boldsymbol{u}_i'\boldsymbol{x}_j) \qquad (2.15)\end{aligned}$$

Then, under the relax and split structure with the auxiliary variable $\mathbf{V} = \mathbf{UX}$, we will get a relaxed problem with the following expression:

$$\arg\min_{\mathbf{V}^{(k)}} \sum_{j=1}^{k}\sum_{i=1}^{m} -G(v_{ij}) + \frac{1}{2\nu}\|v_{ij} - (\mathbf{UX})_{ij}\|_F^2, \text{ or} \qquad (2.16)$$

$$\arg\max_{\mathbf{V}^{(k)}} \sum_{j=1}^{k}\sum_{i=1}^{m} \ G(v_{ij}) - \frac{1}{2\nu}\|v_{ij} - (\mathbf{UX})_{ij}\|_F^2 \qquad (2.17)$$

where $G(v_{ij}) = \log \ f_{\text{Laplace}}(v_{ij})$ is the log likelihood of Laplace density data. To solve this problem, we can consider its sub-problem:

$$\arg\max_{v_{ij}} \ G(v_{ij}) - \frac{1}{2\nu}\left(v_{ij} - (\mathbf{UX})_{ij}\right)^2 \qquad (2.18)$$

And now we can get to its closed form solution with the following steps:

$$\begin{aligned}&\text{Take derivative: } G'(v_{ij}) - \frac{1}{\nu}(v_{ij} - (\mathbf{UX})_{ij}) \overset{set}{=} 0 \\ &\text{Log-likelihood: } G'(v_{ij}) = \frac{\partial \ \log \ f_{\text{Laplace}}(v_{ij})}{\partial \ v_{ij}} = \frac{1}{f(v_{ij})} \cdot \frac{\partial f(v_{ij})}{\partial \ v_{ij}} \\ &\qquad\qquad\qquad\quad = -\frac{|v_{ij}|'}{\lambda} = -\frac{\text{sign}(v_{ij})}{\lambda} \\ &\text{Plug in the } G'\text{: } -\frac{\text{sign}(v_{ij})}{\lambda} - \frac{1}{\nu}(v_{ij} - (\mathbf{UX})_{ij}) \overset{set}{=} 0\end{aligned}$$

It can be simplified to the following equation:

$$v_{ij} + \frac{\nu}{\lambda} \cdot \text{sign}(v_{ij}) = (\mathbf{UX})_{ij} \qquad (2.19)$$

which we should discuss in two situations given $\nu, \lambda > 0$:

$$\begin{cases} 1.\ (\mathbf{UX})_{ij} > 0 : \ \mathrm{sign}(v_{ij}) > 0 \\ 2.\ (\mathbf{UX})_{ij} < 0 : \ \mathrm{sign}(v_{ij}) < 0 \end{cases}$$

Therefore, we can get $\mathrm{sign}((\mathbf{UX})_{ij}) = \mathrm{sign}(v_{ij})$ and solve the problem:

$$\begin{aligned} v_{ij} &= (\mathbf{UX})_{ij} - \frac{\nu}{\lambda} \cdot \mathrm{sign}(v_{ij}) \\ &= |(\mathbf{UX})_{ij}| \cdot \mathrm{sign}((\mathbf{UX})_{ij}) - \frac{\nu}{\lambda} \cdot \mathrm{sign}((\mathbf{UX})_{ij}) \\ &= (|(\mathbf{UX})_{ij}| - \frac{\nu}{\lambda}) \cdot \mathrm{sign}((\mathbf{UX})_{ij}) \end{aligned} \tag{2.20}$$

2.6 Newton's Method for Continuous Functions

When dealing with continuous functions, Newton's method is often applied to find the minimum values, which is computationally friendly and is able to find a converged solution in most cases. As previously discussed, ICA is a problem to minimize the mutual information, or equivalently, to maximize log-likelihood. We know that the question of finding the minimum (or maximum) can be transferred to the question of finding roots of its first derivative when the function is smooth and without boundary. This is the point why we can apply Newton's method here.

In practice, different density functions $f(.)$ can be used to get the log-likelihood function, here we use the logistic density to show the procedure:

$$f(v_{ij}) = \frac{e^{(v_{ij}-\mu)\cdot a}}{-\frac{1}{a} \cdot (1 + e^{(v_{ij}-\mu)\cdot a})^2}\ , (a < 0) \tag{2.21}$$

This logistic density has variance of $\frac{a^2\pi^2}{3}$ and mean of μ. Under the relax and split framework, we still start from the problem of minimizing mutual information, and then arrive at the sub-problem as mentioned in the equation 2.18:

$$\underset{v_{ij}}{\arg\max}\ G(v_{ij}) - \frac{1}{2\nu}\left(v_{ij} - (\mathbf{UX})_{ij}\right)_F^2 \tag{2.22}$$

where $G(v_{ij})$ is the log-likelihood function that needs to be maximized. When we centered the data to have mean zero and unit variance, it will be the following form with $a = -\frac{\pi}{\sqrt{3}}$:

$$G(v_{ij}) = \log\ f(v_{ij}) = a \cdot v_{ij} + log(-a) - 2 \cdot \log\ \left(1 + e^{a \cdot v_{ij}}\right) \tag{2.23}$$

And then, we can follow these steps to apply Newton's method:

Original equation: $\underset{v_{ij}}{\arg\max}\ G(v_{ij}) - \frac{1}{2\nu}\|v_{ij} - (\mathbf{UX})_{ij}\|_F^2$

First derivative: $G' - \frac{1}{\nu}\left(v_{ij} - (\mathbf{UX})_{ij}\right) = a - \frac{2a \cdot e^{v_{ij} \cdot a}}{1 + e^{v_{ij} \cdot a}} - \frac{1}{\nu}\left(v_{ij} - (\mathbf{UX})_{ij}\right)$

Define Z(v_{ij}): $Z(v_{ij}) = \frac{2a \cdot \nu \cdot e^{v_{ij} \cdot a}}{1 + e^{v_{ij} \cdot a}} + \left(v_{ij} - (\mathbf{UX})_{ij}\right) - a \cdot \nu$

Take the derivative: $Z'(v_{ij}) = \frac{4a^2 \cdot \nu \cdot e^{v_{ij} \cdot a}}{(1 + e^{v_{ij} \cdot a})^2} + 1$

Newton's Method: $v_{n+1} = v_n - \frac{Z(v_n)}{Z'(v_n)}, (n = 0, 1, 2, ...)$

(Stop gradient updating when V is converged.)

2.7 Orthogonal Procrustes Problem

As for the problem of updating matrix **U** in the relax and split optimization:

$$\underset{\mathbf{U}^T\mathbf{U}=\mathbf{I}}{\arg\min} \|\mathbf{V} - \mathbf{UX}\|_F^2, s.t.\mathbf{U}^T\mathbf{U} = I \tag{2.24}$$

We first expand the quadratic term to derive its closed form solution:

$$\begin{aligned}
\|\mathbf{V}-\mathbf{U}\mathbf{X}\|_F^2 &= \sum_{i,j}(\mathbf{V}-\mathbf{U}\mathbf{X})_{i,j}^2 \\
&= \sum_{i,j}(\mathbf{V})_{i,j}^2 + (\mathbf{U}\mathbf{X})_{i,j}^2 - 2(\mathbf{V})_{i,j}(\mathbf{U}\mathbf{X})_{i,j} \\
&= \|\mathbf{V}\|_F^2 + \|\mathbf{U}\mathbf{X}\|_F^2 - 2tr(\mathbf{V}^T\mathbf{U}\mathbf{X}) \\
&= \|\mathbf{V}\|_F^2 + \|\mathbf{X}\|_F^2 - 2tr(\mathbf{X}\mathbf{V}^T\mathbf{U})
\end{aligned}$$

Thus, the problem of updating matrix $\mathbf{U}$ is equivalent to maximize $tr(\mathbf{X}\mathbf{V}^T\mathbf{U})$. Let $\tilde{\mathbf{U}}\tilde{\Sigma}\tilde{\mathbf{V}}^T$ be the singular value decomposition of $\mathbf{X}\mathbf{V}^T$, we then have:

$$\begin{aligned}
tr(\mathbf{X}\mathbf{V}^T\mathbf{U}) &= tr(\tilde{\mathbf{U}}\tilde{\Sigma}\tilde{\mathbf{V}}^T\mathbf{U}) \\
&= tr(\tilde{\mathbf{V}}^T\mathbf{U}\tilde{\mathbf{U}}\tilde{\Sigma})
\end{aligned}$$

Let $\mathbf{C} = \tilde{\mathbf{V}}^T\mathbf{U}\tilde{\mathbf{U}}$, then it will also be orthogonal as the product of orthogonal matrices. Since $\tilde{\Sigma}$ is a diagonal matrix, we can have:

$$tr(\mathbf{C}\tilde{\Sigma}) = \sum_{i=1}^{n} C_{ii} \cdot \tilde{\Sigma}_{ii},$$

where C_{ii} are non-negative values. To maximize this equation under the orthogonal assumption, we will need to make the $C_{ii} = 1$ so that:

$$\begin{aligned}
&\mathbf{C} = \tilde{\mathbf{V}}^T\mathbf{U}\tilde{\mathbf{U}} = \mathbf{I}_n, \\
&\mathbf{U} = \tilde{\mathbf{V}}\tilde{\mathbf{U}}^T \qquad (2.25)
\end{aligned}$$

2.8 Simulation Study

We simulated data under the structure of LNGCA model with three true signals (m = 3) out of a total of 50 components. The true signals can be plotted as 33×33 images, which when vectorized corresponds to k = 1089 observations for each component, and the "active" pixels will show in the shape of "1", "2 2", or "3 3 3". Four algorithms were tested to compare their performance of recovering independent components:

1. Relax and split approximation using Laplace density (Relax-Laplace);
2. Relax and split approximation using logistic density (Relax-logistic);
3. FastICA using log cosh function (FastICA-tanh);
4. FastICA using logistic function (FastICA-logistic).

To measure the accuracy of each ICA method, we used the sign- and permutation-invariant mean-squared error (PMSE) to represent the minimum distance (d_{MD}) as Risk used in their linear non-Gaussian component analysis [8]. Let $\hat{\mathbf{W}}$ denote the estimated unmixing matrix derived by four methods, and $\mathbf{W}$ denote the true unmixing matrix, we examine $d_{MD}(\hat{\mathbf{W}}, \mathbf{W})$ to check the accuracy and the consistency of results.

Given different initial values, we simulated 250 times for each algorithm. Different scale parameters of ν were also examined in order to determine the effect of the sparsity-controlling parameter on non-convexity.

Another measurement of accuracy is the probability of getting the first success that recover all the three independent components, in a pool of 50 initialization values. Geometric distribution is applied to calculate the probability. After repeating 1000 times (i.e. 50 initialization each time), we can calculate the average probability for these 1000 experiments as well as its confidence interval, which then can be used to evaluate accuracy and consistency for each algorithm.

2.9 Real Data Application

The real data used in this study are the 25 principal components from the 1200-subjects HCP data release (March 2017), containing 1003 subjects with four complete rfMRI runs (4800 total timepoints). Data were preprocessed by the HCP as described in the minimal preprocessing pipelines [5]. This resulted in data in grayordinate space, which includes approximately 60,000 cortical vertices and 30,000 subcortical voxels. Data was temporally demeaned and variance normalized. Melodic Incremental Group PCA was conducted and the top 25 components retained. We then conducted ICA on these components. This is equivalent to conducting group ICA[3].

To check the consistency of ICA methods, we run 50 times with different initialization values here, and select the estimator with the minimum mutual information (or the maximum non-Gaussianity) as the best estimator, then calculate the distance between each estimator and the best estimator.

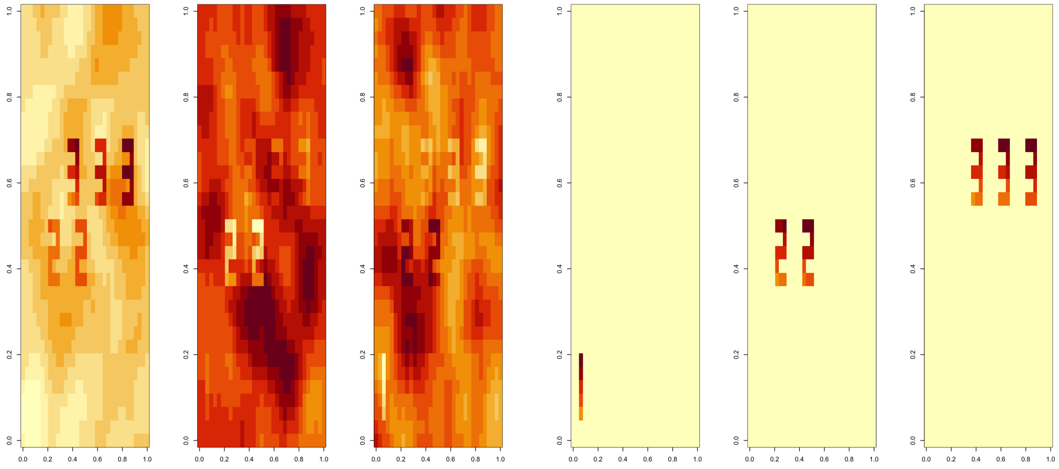
Connectome Workbench (v1.4.2) is used to visualize the ICA maps and compare the difference among the best estimators that generated by each method.

Chapter 3

Results

3.1 Tuning Scale Parameters

The simulated data contain both the observed data matrix (**X**) and the true signal matrix (**S**). Using the LNGCA models, we specified 3 signal components out of total 50 components, so that the other 47 ones are defined as noise. The following figures visualize the data as images.



(a) Simulated **X** matrix

(b) Simulated **S** matrix

Figure 3.1: Visualization of simulated data in 33 × 33 images

For the method of Relax-Laplace, we tested the scale parameter ν of being 0.1, 0.5, 1 and 2. With a larger parameter of ν, the resulting algorithm tends to be less accurate (Figure 3.2). A smaller ν might produce more accurate and consistent results but it uses more computation time. Meanwhile, with a smaller ν, the results will be less sparse, which should also be taken into consideration when applying to fMRI data.

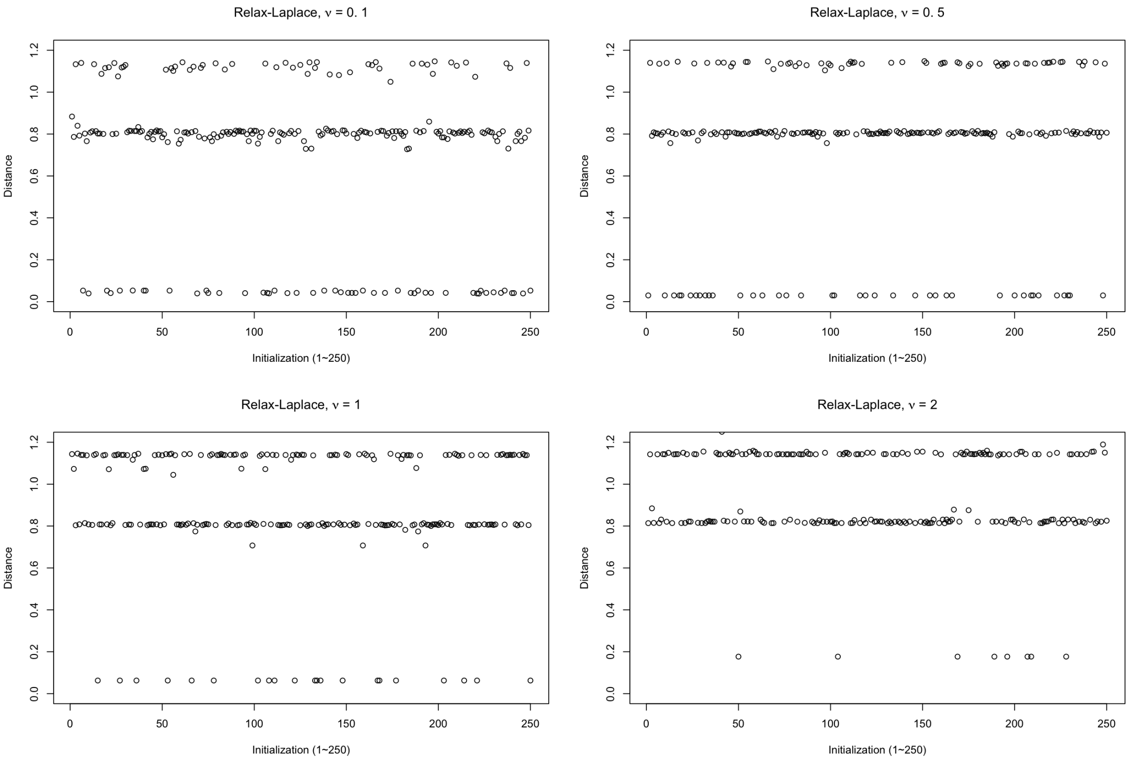


Figure 3.2: Relax-Laplace method with different scale parameters (ν)

For the method of Relax-logistic, we tested the scale parameter ν of being 0.1, 0.5, 1 and 2. Similar to the situation of Relax-Laplace, with a larger parameter of ν, Relax-logistic will be less accurate, a smaller ν will produce more accurate and consistent results but consume more computation time (Figure 3.3).

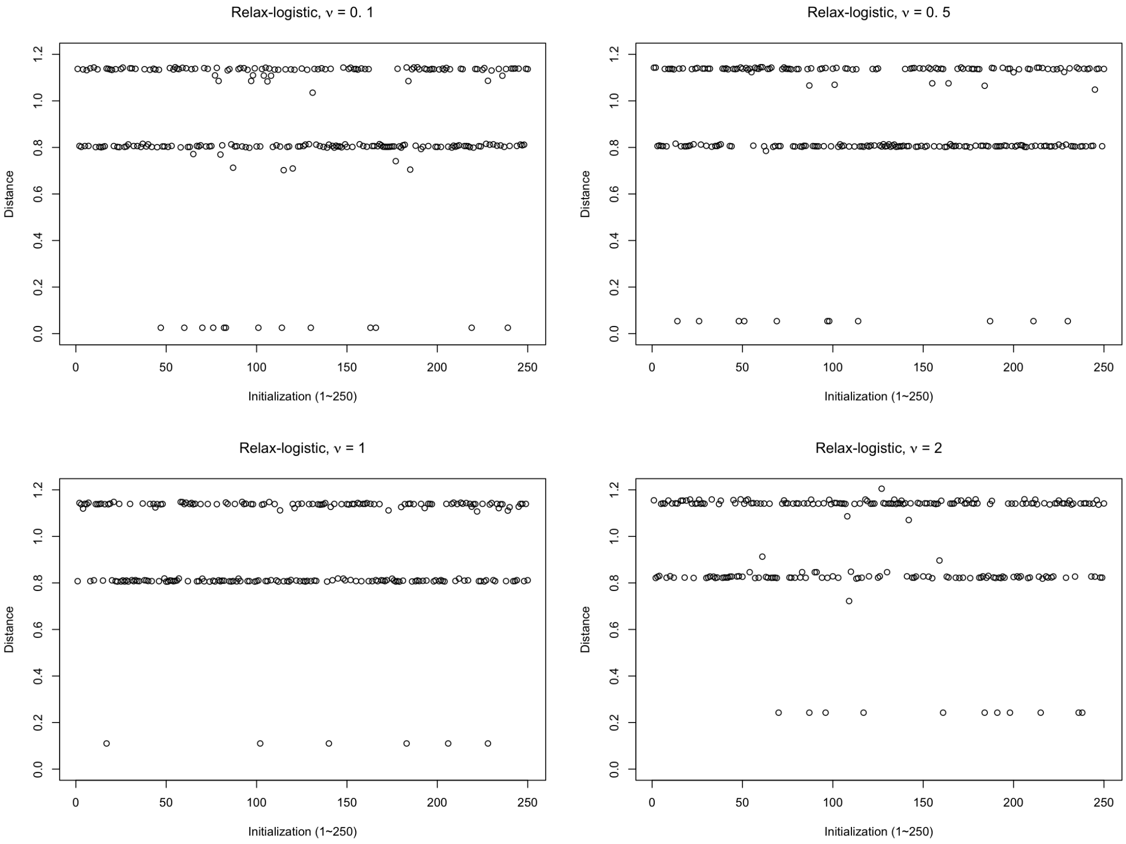


Figure 3.3: Relax-logistic method with different scale parameters (ν)

For the method of Relax-Laplace, we choose $\nu = 1$ with the consideration of accuracy and also the sparsity of the outputs.

For the method of Relax-logistic, We also tested for $\nu = 0.01$. It can still increase the accuracy but consumes much more time to find a converged solution. Thus, we choose $\nu = 0.1$ with the consideration of accuracy and computation time.

3.2 Convergence and Accuracy of the 4 Methods

Using the optimized parameter ν for the method of Relax-Laplace and the method of Relaxed-logistic, we did simulation with 250 times for all the four methods (Figure 3.4).

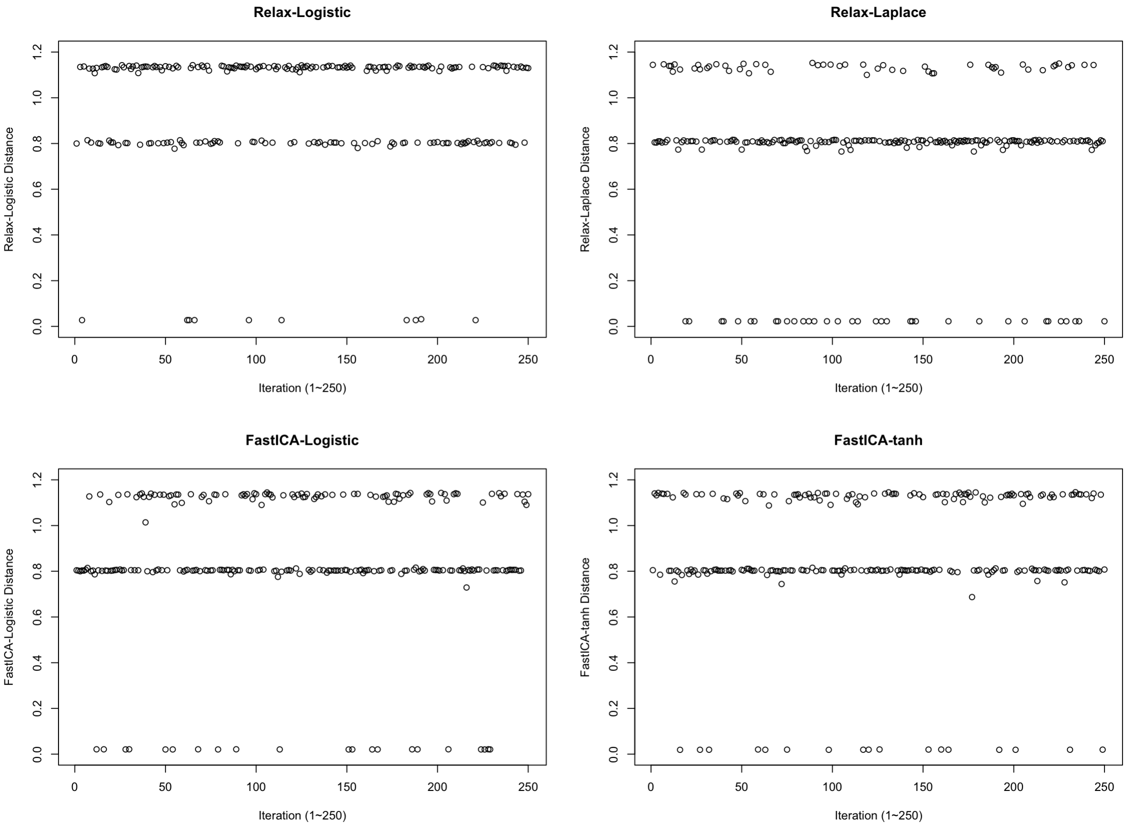


Figure 3.4: Simulation results with 250 iterations

From the above figures, we learned that the Relax-Laplace are more accurate under the current scale parameters. The Relax-logistic as well as FastICA methods will have a higher chance to fail recovering all three true components.

3.3 The First Success Probability of the 4 Methods

The first success is defined as recovering all the 3 components in the simulation data, namely the minimum distance (d_{MD}) is approaching zero. Here, we used the criterion of "less than 0.1" to decide the successful results based on the convergence situations found in Figure 3.

During our simulation, a pool of 50 initialization was used in each iteration, and a total of 1000 iterations were repeated. Geometric distribution was then used to calculate the probability of getting the first success (Table 3.1).

	Method	**N**	**Probability**	**95% CI**
1	Relax-logistic	989	0.048	0.048 ± 0.001
2	Relax-Softmax	999	0.084	0.084 ± 0.002
3	FastICA-logistic	985	0.051	0.051 ± 0.001
4	FastICA-tanh	992	0.056	0.049 ± 0.001

Table 3.1: The probability of getting the first success within 50 initializations

The method of Relax-Laplace turned out to have the highest probability among the four methods, the other three were similar to each other. All of the variance is small, indicating the estimation is consistent across the 1000 times simulation.

Taking the average of probabilities from the table, we can generate a figure of cumulative mass function to show the four methods (Figure 3.5).

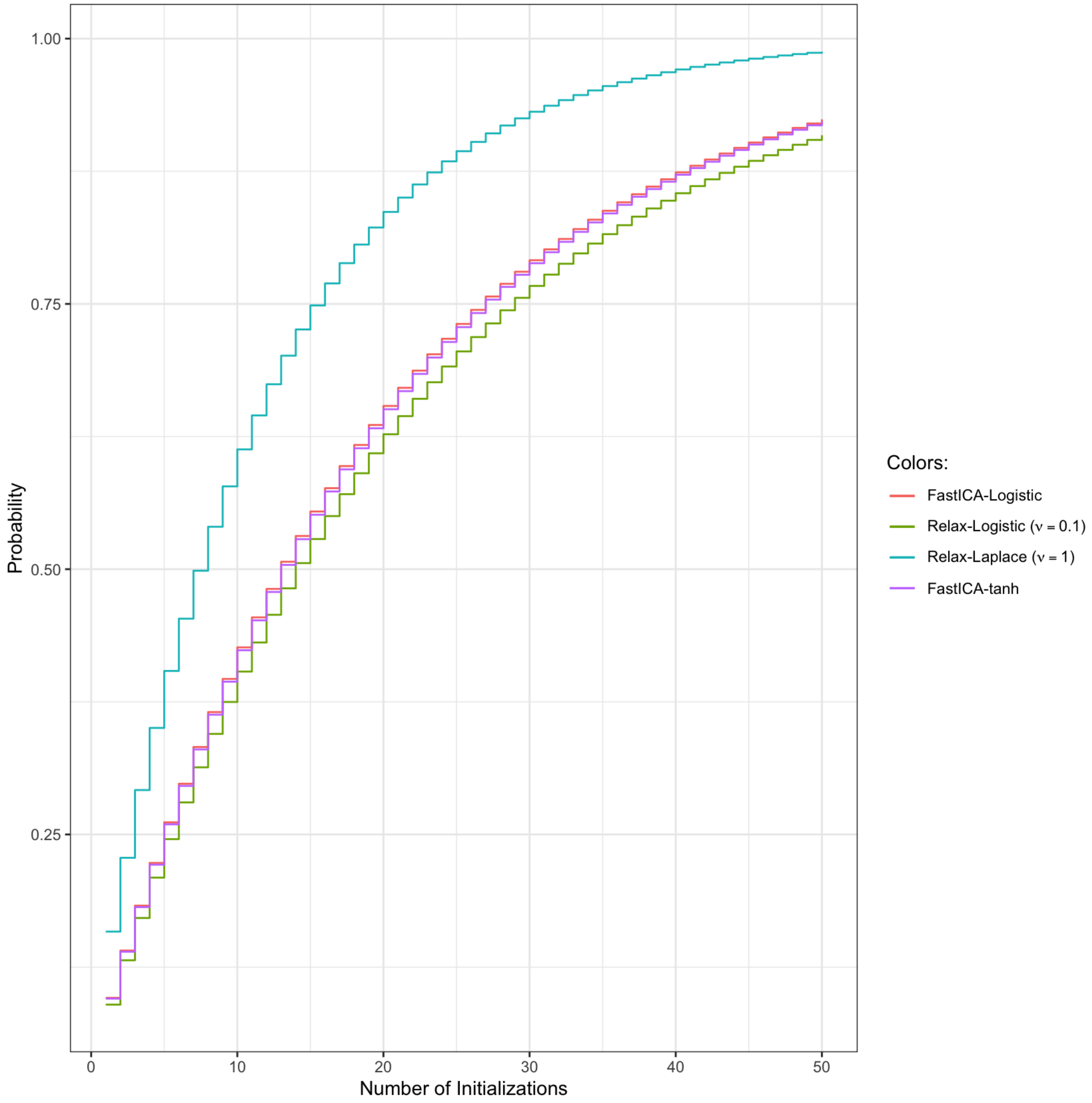


Figure 3.5: Cumulative mass functions of 4 methods using average probabilities

From this figure, we can get the same results as the Table 3.1. The Relax-Laplace method will have a greater probability to achieve the first success given a same number of try.

3.4 Applications on the Real Data

We also applied the methods on the real data from 1003 subjects made public by the Human Connectome Project. Data matrix has the dimension of $\mathbf{X} \in \mathbb{R}^{91282 \times 25}$, and the method of Relax-Laplace with $\nu = 1$ was used to produce sparse signals. FastICA-logistic and FastICA-tanh were also applied to compare the consistency and accuracy. However, we didn't apply Relax-logistic on the real data since it would produce similar accuracy like the FastICAs under $\nu = 0.1$, and it consumes more time if we use $\nu = 0.01$ for a better performance.

Based on the log likelihood values, we selected the best estimator from a pool of 50 estimators, then calculated the minimum distance d_{MD} between each estimator and the best one (Figure 3.6).

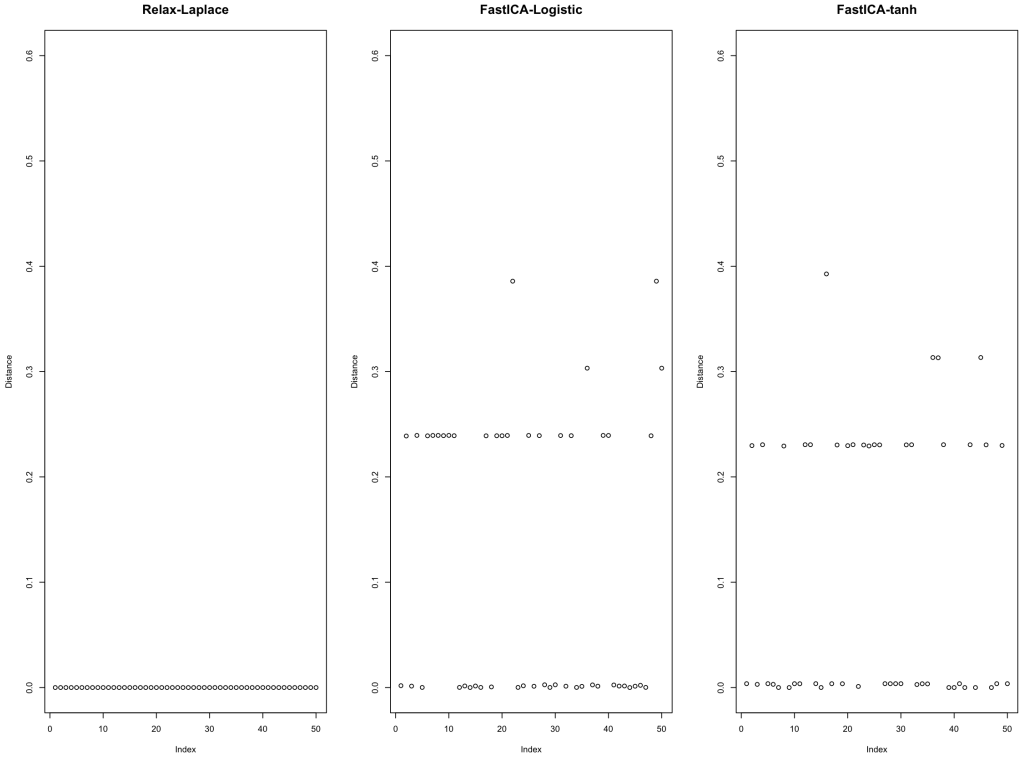


Figure 3.6: Minimum Distances from the best estimator

Using the criterion of $d_{MD} < 0.1$, we can calculate the probability of matching the best estimator: Relax-Laplace = 1, FastICA-logistic = 0.54, FastICA-tanh = 0.56. Therefore, we can make a plot of the cumulative mass function under a geometric distribution (Figure 3.7).

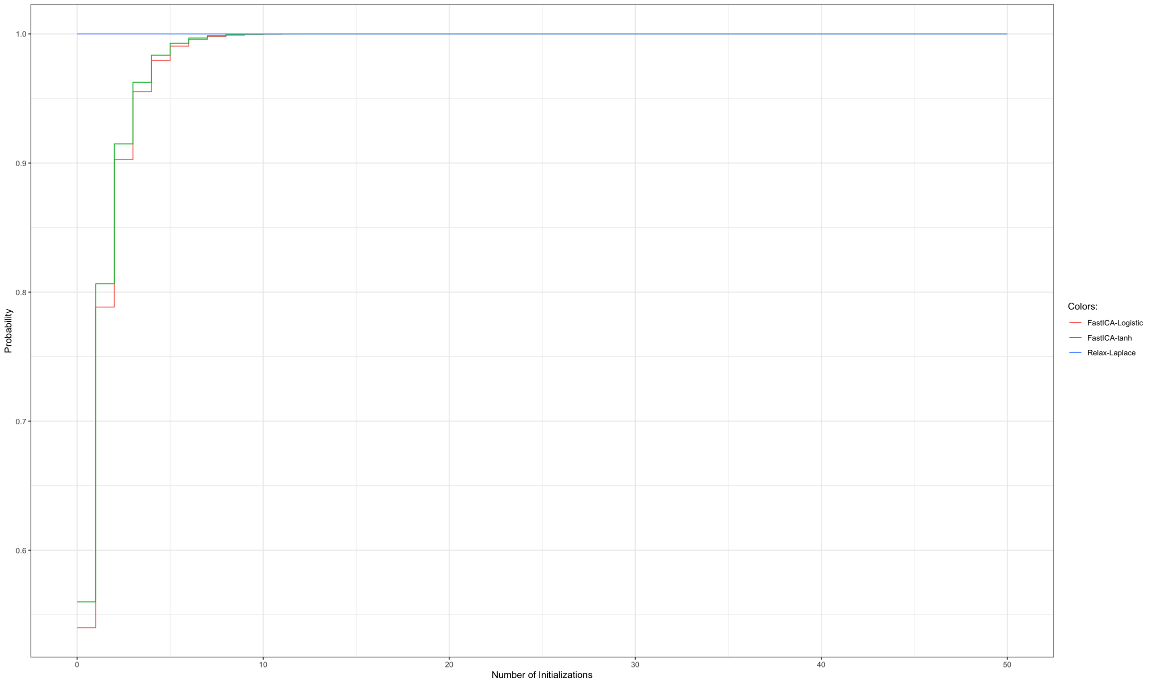


Figure 3.7: Cumulative mass functions using the probability of matching the best estimator in HCP real data application

To visualize the components, we used the Connectome Workbench to map the data on a brain surface. The following figure shows the difference between FastICA methods and Relax-Laplace method. Since the two FastICA methods produced similar results, here we only presented the comparisons between FastICA-tanh and Relax-Laplace (Figure 3.8).

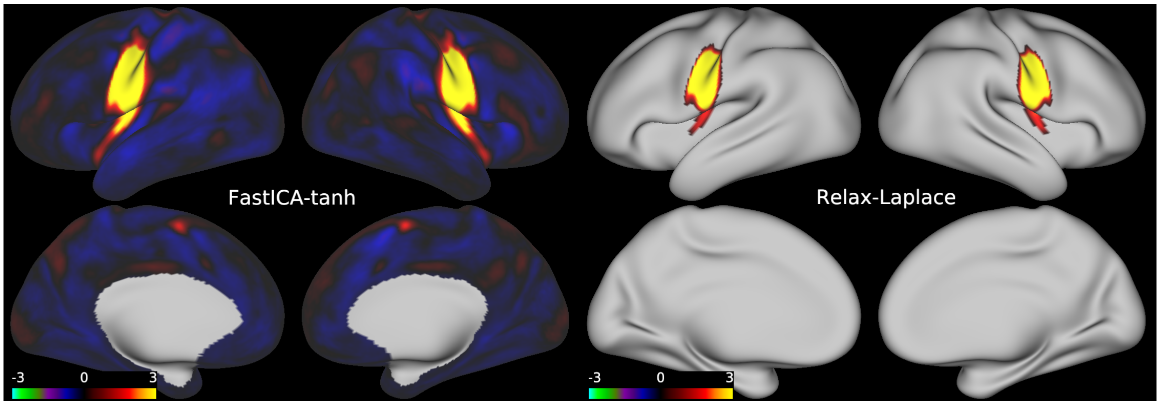


Figure 3.8: Motor Cortex Visualization: FastICA-tanh (Left) vs Relax-Laplace (Right)

Similarly, we can show the visualize of the default mode network (Figure 3.9).

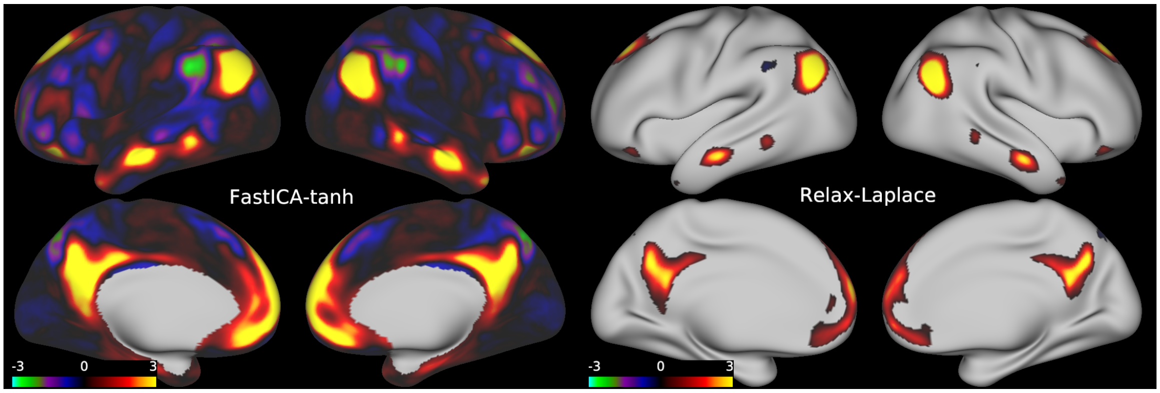


Figure 3.9: Default Mode Network Visualization: FastICA-tanh (Left) vs Relax-Laplace (Right)

The FastICAs produced signals everywhere on the brain, whereas the Relax-Laplace produced sparse signals under the Laplace density. We can control the scale of sparsity by changing the parameter of ν.

Chapter 4

Discussion

Relax and split optimization used in this study allows the optimization of a non-convex objective function, which arises due to orthogonality constraints. It also allows the optimization of non-smooth objective functions, like the Laplace density, which can be used to encourage sparsity.

Relax-Laplace has a simple coding structure that can be applied to large data sets, is computationally efficient, and results in sparse matrices. We can also change the scale parameter to control the sparsity of its outcomes.

Another thing to note for Relax-Laplace method is the tuning scale parameter ν. In this study, we only conducted a grid search on $\nu = 0.1$, 0.5, 1 and 2. However, the result shows that being smaller does not always improve the performance of Relax-Laplace. For example, when applying on the real data, the consistency of estimators with $\nu = 1$ is better than estimators with $\nu = 0.5$. Further evaluations can be made to try different scale parameters and increase the experiment size.

Relax-logistic method used in this study has met more problems with its scale parameter. From 0.01 to 2, we found that a smaller ν can always improve its accuracy and consistency. However, using $\nu = 0.01$ consumes much more time than Relax-Laplace and the two FastICA methods, which therefore limited its application on large data sets. Of course, we are not restricted on the logistic density, other smooth functions are also possible to use the relax and split optimization with the approximation of Newton's method.

Bibliography

[1] Beckmann, C. F. (2012). Modelling with independent components. *Neuroimage*, 62(2):891–901.

[2] Biswal, B. B., Mennes, M., Zuo, X.-N., Gohel, S., Kelly, C., Smith, S. M., Beckmann, C. F., Adelstein, J. S., Buckner, R. L., Colcombe, S., et al. (2010). Toward discovery science of human brain function. *Proceedings of the National Academy of Sciences*, 107(10):4734–4739.

[3] Calhoun, V. D., Adali, T., Pearlson, G. D., and Pekar, J. J. (2001). A method for making group inferences from functional mri data using independent component analysis. *Human brain mapping*, 14(3):140–151.

[4] Cardoso, J.-F. (1998). Blind signal separation: statistical principles. *Proceedings of the IEEE*, 86(10):2009–2025.

[5] Glasser, M. F., Sotiropoulos, S. N., Wilson, J. A., Coalson, T. S., Fischl, B., Andersson, J. L., Xu, J., Jbabdi, S., Webster, M., Polimeni, J. R., et al. (2013). The minimal preprocessing pipelines for the human connectome project. *Neuroimage*, 80:105–124.

[6] Hyvarinen, A. (1999). Fast and robust fixed-point algorithms for independent component analysis. *IEEE transactions on Neural Networks*, 10(3):626–634.

[7] Hyvärinen, A. and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5):411–430.

[8] Risk, B. B., Matteson, D. S., and Ruppert, D. (2019). Linear non-gaussian component analysis via maximum likelihood. *Journal of the American Statistical Association*, 114(525):332–343.

[9] Risk, B. B., Matteson, D. S., Ruppert, D., Eloyan, A., and Caffo, B. S. (2014). An evaluation of independent component analyses with an application to resting-state fmri. *Biometrics*, 70(1):224–236.

[10] Zheng, P. and Aravkin, A. (2018). Fast methods for nonsmooth nonconvex minimization. *arXiv preprint arXiv:1802.02654*.

[11] Zheng, P., Risk, B., Gaynanova, I., and Aravkin, A. (2019). Relax and split algorithm for ICA. In *ENAR Spring Meeting, Philadelphia, PA*.